

Harold Fleisher
Morton Tavel
John Yeager

Exclusive-OR Representations of Boolean Functions

With the goal of making exclusive-OR formulations of switching functions more readily available to designers for implementation in LSI and VLSI technologies, we introduce the concept of an exclusive-OR space in which an exclusive-OR normal form is defined to correspond to the conventional disjunctive normal form. A geometrical representation of exclusive-OR space is described, and its various bases are listed and discussed.

Introduction

Many useful logical functions have a high degree of exclusive-OR content (counters and parity functions, for example), and one would expect them to exhibit a particular economy of form when expressed as exclusive-OR sums rather than in the more conventional disjunctive normal form. In addition, it has been shown that switching functions are more easily testable when expressed as exclusive-OR sums using the Reed-Muller canonical form [1-3], so that this realization can be important in the design of fault tolerant computers. Expressing such functions as minimal exclusive-OR sums is quite important when they are to be implemented with LSI and VLSI technology, as, for example, in appropriately designed Programmable Logic Arrays (PLAs) or cellular logic arrays [4]. The problem of minimizing exclusive-OR sums is not a simple one, however, and when the number of variables is large or when certain variables must appear both in complemented and uncomplemented form, it appears that only heuristic methods, which do not guarantee global minimization, are possible [4-6].

Although the literature on modulo-two functions is extensive (see, for example, Davio et al. [7]), discussions of their representations and minimization methods are algebraic and exhibit a high degree of mathematical sophistication. We have found that very often geometrical representations are easier to comprehend than algebraic ones and, because of their more intuitive nature, can offer insights that may be more useful for engineering applications. The purpose of this

paper, therefore, is to present a geometrical construction that can be used to represent exclusive-OR functions in the same way that the usual Boolean cube is used to represent ordinary OR sums. We show how this exclusive-OR cube elucidates the use of various bases that are available for exclusive-OR representations, including bases in which variables can be present in either all complemented, all uncomplemented, or mixed form. It is our hope that this representation may help to clarify the manipulation of logical functions, including their minimization, and perhaps be suggestive of more potent algebraic methods.

Exclusive-OR space

The minterms of the disjunctive normal form (DNF) provide a basis for the space of Boolean functions expressed as polynomials in the form of an OR-sum ($\vee$) of product ($\wedge$) terms. Analogously, one can define a set of terms to serve as a basis of the vector space of Boolean functions expressed as polynomials with an exclusive-OR (∇) sum [8]. We call these terms "onterms." The onterms are those products of the logical variables in which no variable appears in complemented form. The word "basis" is used here in its strict mathematical sense, to denote a set of linearly independent elements which span a space. The exclusive-OR space is a true vector space, in that the operations $\wedge$ and ∇ are in fact the modulo-two product and sum which comprise a field. This is not true of the disjunction space, since the pair $\wedge$ and $\vee$ do not define a field, as neither is a group operation. A

© Copyright 1983 by International Business Machines Corporation. Copying in printed form for private use is permitted without payment of royalty provided that (1) each reproduction is done without alteration and (2) the *Journal* reference and IBM copyright notice are included on the first page. The title and abstract, but no other portions, of this paper may be copied or distributed royalty free without further permission by computer-based and other information-service systems. Permission to *republish* any other portion of this paper must be obtained from the Editor.

function expressed in the basis of onterms is said to be in exclusive-OR normal form (ENF). Thus, the ENF of the function $\bar{a} \vee ab$ is given by

$$\bar{a} \vee ab = 1 \vee a \vee ab.$$

The terms 1, a , and ab are onterms. As we have indicated, we call the space spanned by this basis the exclusive-OR space, or simply the E-space, in contrast to the disjunction space (D-space) spanned by the minterms of the DNF. Many functions, such as the parity function, are more naturally expressed in E-space, because they are either of a counting or comparing nature.

These two spaces of functions of N variables may be represented geometrically as N -cubes, hypercubes in N -dimensional Euclidean space. While the use of such cubes for D-space representation is widespread [9], the E-space cube to be discussed here appears to be a new construction. Three-dimensional examples of the D- and E-cubes are illustrated in Fig. 1. The vertices of the E-cube are the onterms. In both cubes, the higher-dimension subcubes represent the appropriate sum of their vertices. Thus, on the E-cube, the square with vertices 1, a , b , and ab represents the term $1 \vee a \vee b \vee ab$; i.e., the term $\bar{a}\bar{b}$. The number of complemented variables in a term represented by a subcube of the E-cube is equal to the dimension of that subcube. To represent a function as a sum of subcubes, we may think of each subcube as possessing a binary value (0 or 1) determining whether its term appears in the polynomial. If a subcube has a value of 1, its term is present in the polynomial; a value of 0 means it is not.

It should be noted that any p -dimensional subcube may have its value complemented by complementing any of the p pairs of parallel $(p - 1)$ -dimensional subcubes it contains. Thus, any p -dimensional subcube may be replaced by any of these pairs. A line may be replaced by the pair of points that bound it, a square by either of the two pairs of parallel lines it contains, and a tesseract (4-cube) may be replaced by any of the four pairs of parallel cubes it contains. Several well-known devices in symbolic logic are actually representations of these cubes. An example is the Karnaugh map, both in its standard form [10] and as applied to exclusive-OR polynomials in [11].

One major difference exists between the D- and E-cubes. When a function is expressed on the D-cube, any vertex contained in more than one of the subcubes composing the function (as in the case of the vertex abc of the function $ac \vee bc$) is simply assigned a value of 1 when the function is projected into the minterm basis. In the case of the E-cube, however, such an overlap may cause a cancellation of the vertex. As an example of this, the function $a\bar{b} \vee \bar{a}b$ (two intersecting lines on the E-cube) does not contain the vertex

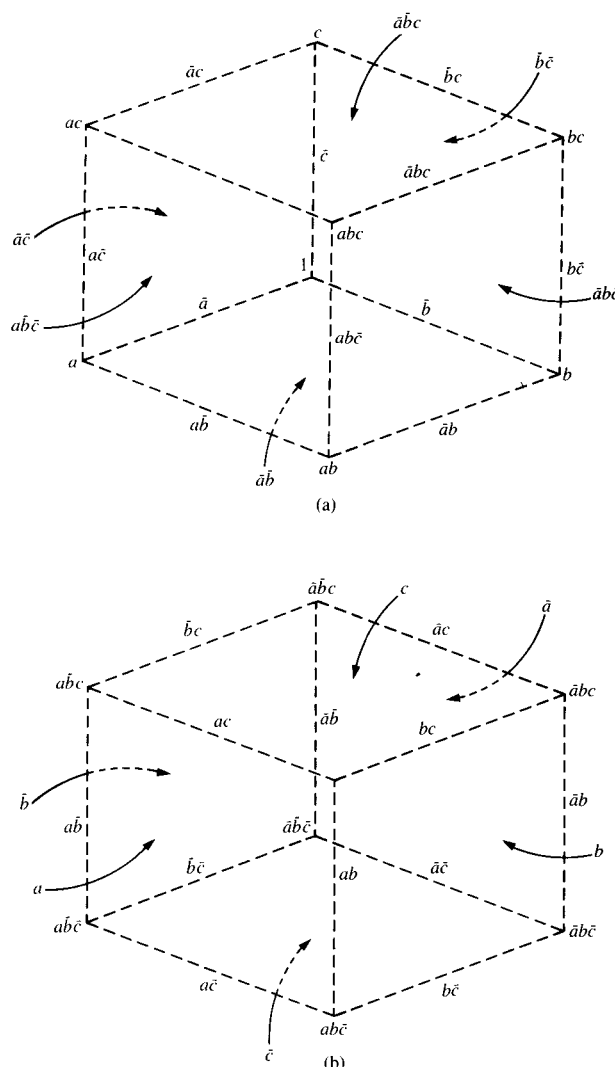


Figure 1 (a) The E-cube; the entire cube represents the term abc . (b) The D-cube; the entire cube represents the term 1.

ab , even though this vertex is contained in each line; i.e., $a\bar{b} \vee \bar{a}b = a \vee b$. Only when an odd number of subcubes containing the vertex have a value of 1 will that vertex have the value 1 also. It is precisely this feature which provides the richer structure of the E-cube and makes it interesting, both in terms of variety of expressions for a single function and the number of distinct bases of the space. E-space, in fact, has several distinct types of bases, while D-space has only one. In D-space, if any term in a sum corresponds to a subcube containing a certain vertex, that vertex must be present and have the value of 1 in the result. Since there can be no cancellation of vertices in D-space and since there exist the 2^N functions which represent the individual minterms, any

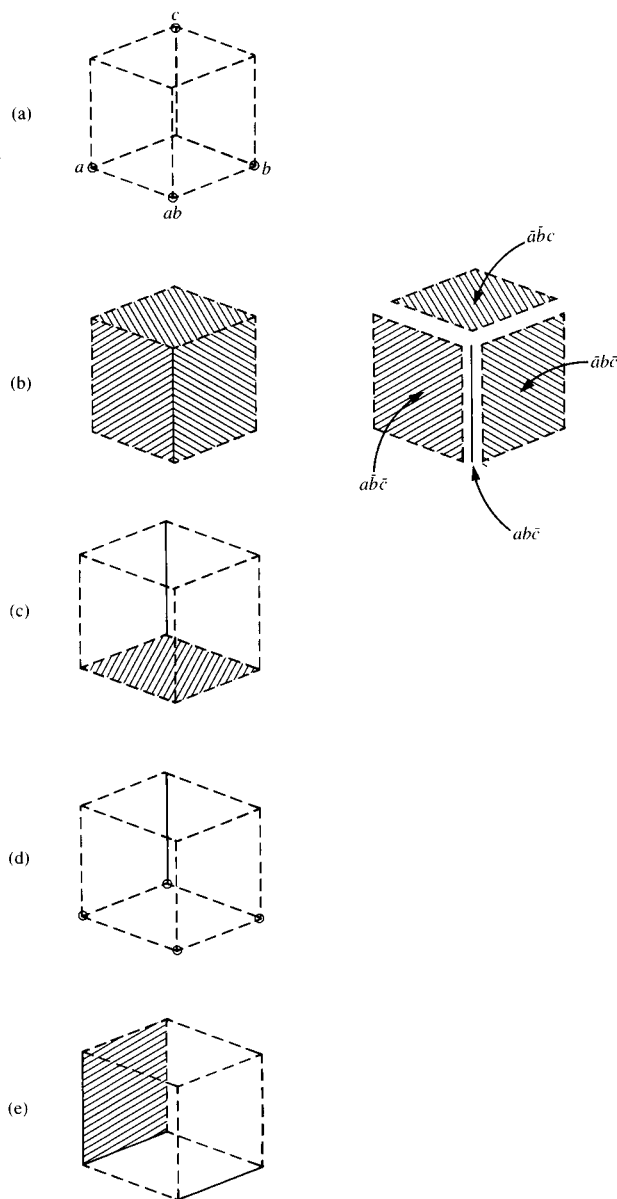


Figure 2 Various bases in E-space with the function $a \vee b \vee c \vee ab$ expressed in each of them. All vertices are labeled as in Fig. 1(a). (a) The ENF basis. The function $a \vee b \vee c \vee ab$ is the exclusive-OR sum of the indicated vertices (ontersms). (b) The DNF basis in E-space. The elements of this basis are the minterms. The function $a \vee b \vee c \vee ab$ is written as $abc \vee abc \vee abc \vee abc$ in this basis, the exclusive-OR sum of three sides and an edge. (c) The basis, CENF, is composed of all subcubes containing the vertex 1. In this basis the function $a \vee b \vee c \vee ab$ is written $\bar{a}\bar{b} \vee \bar{c}$, an edge and the bottom face. (d) A MNF basis. The variables are of classes enf in a , enf in b , and cenf in c . The function $a \vee b \vee c \vee ab$ is written $1 \vee a \vee b \vee ab \vee \bar{c}$, an edge and four vertices. (e) An HNF basis. The variables are of classes dnf in a , enf in b , and cenf in c . The function $a \vee b \vee c \vee ab$ is written $\bar{a} \vee \bar{a}b \vee \bar{a}c \vee ac$, three edges and a face.

basis for D-space must contain the single vertices; i.e., it must have the minterms as a subset. Therefore, the vertices provide the only basis for the D-cube.

In addition to the ENF basis, however, E-space is also spanned by the minterms [8]. In E-space, of course, the minterms are not vertices, but are the set of all subcubes that contain the vertex having all variables present (i.e., the vertex abc in the case of the 3-cube). We call this basis the DNF basis in E-space. The DNF basis is very useful, since all of the minterms are disjoint. If f and g are minterms, then $f \vee g$ is precisely the same function as $f \vee g$. The coefficients of the minterms in disjunction space for an arbitrary function are, therefore, also the coefficients of the elements of this basis in E-space. This provides a useful method for transferring a function from its representation on the D-cube to a representation on the E-cube. In a similar manner, one finds that another basis in E-space is the set of all subcubes that contain the vertex 1. These subcubes are represented by terms that contain only complemented variables, so we call them the "offterms." The basis is referred to as the complemented exclusive-OR normal form (CENF). Figure 2 gives examples of five different bases for E-space and shows how the function $a \vee b \vee c \vee ab$ is represented in each basis.

We shall now catalog the 3^N bases for the E-space. We call these bases the hybrid normal forms (HNFs). They can be considered as combinations of the three "pure" bases, ENF, CENF, and DNF. Within these hybrid bases, each variable of the product terms must fall into one of the following three classes:

- dnf The class of variables which occur in all the terms (either complemented or uncomplemented).
- enf The class of variables which never occur complemented in any term.
- cenf The class of variables which never occur uncomplemented in any term.

It should be noted that an arbitrary function in E-space may contain variables that are not members of any of these classes; these classes are intended to classify all the variables in a given basis. The subset of the HNF bases consisting of those in which all variables are either class enf or cenf are called mixed normal forms (MNFs). There are 2^N such MNFs. Figure 2(d) illustrates one such MNF and shows how it is used to represent the function $a \vee b \vee c \vee ab$.

The process of expressing a function in a given basis is quite simple and can be thought of in terms of operators symbolized: $\langle \text{basis} \rangle$. We shall consider here only the operator $\langle \text{ENF} \rangle$, which projects a function into the ENF basis. The operators projecting into other bases are quite similar. Consider a function corresponding to a certain set of subcubes of the E-space cube. Each logical variable which occurs in complemented form, and thus cannot be a class enf, will be replaced by 1 exclusive-ORed with that variable. Thus, the variable $\bar{c}$ becomes $1 \vee c$. Using the distributive

law, the resulting function can be converted to a set of onterms. For example, the function $\bar{a} \wedge \bar{b}$ will become $(1 \vee a) \wedge (1 \vee b)$, which finally becomes $1 \vee a \vee b \vee ab$. Any duplicate terms created should be canceled (modulo two).

A very useful result can be obtained by viewing every MNF as an ENF with a simple change in variables (i.e., wherein all variables that are of the class cenf have been complemented). For example, the function $b \vee \bar{a}$ is in ENF if $\bar{a}$ is considered to replace a as a variable (a becoming the complement of the new variable $\bar{a}$). Since the subcubes of the DNF basis are precisely those which are symmetric to the CENF terms with respect to the center of the cube, similar transformations from DNF to ENF and from DNF to CENF are implied.

An operator that projects arbitrary functions into a given HNF basis may be written as the product of operators, each of which sends a given variable into its designated class. For example, consider the basis wherein a and d are class cenf, b is of class enf, and c and e are of class dnf. If by $\langle enf a \rangle$ we denote the operation which sends an arbitrary set of subcubes into a set in which the variable a is of class enf and the other variables are unchanged, then the operator which projects into HNF can be written

$$\langle HNF \rangle = \langle cenf a \rangle \langle enf b \rangle \langle dnf c \rangle \langle cenf d \rangle \langle dnf e \rangle.$$

These three types of single variable class transformations correspond to the A, B, and C operators introduced by Muller [12].

Consider now the three transformations (in permutation notation):

$$Id, (1 a \bar{a}), \text{ and } (1 \bar{a} a),$$

where Id is the identity transformation; the symbol 1 indicates the absence of a variable in a term; and the operation $(1 a \bar{a})$ means "replace the absent variable with the variable, replace the variable with its complement, and replace the complemented variable with 1." With this notation, the operator $\langle cenf a \rangle$ can be expressed as $(1 a \bar{a}) \langle enf a \rangle (1 \bar{a} a)$, the operator $\langle enf a \rangle$ can be expressed as $(1 \bar{a} a) \langle enf a \rangle (1 a \bar{a})$, and, obviously, $\langle dnf a \rangle$ can be expressed as $Id \langle enf a \rangle Id$. These are similarity transformations of the $\langle enf \rangle$ operator. A complete HNF operator can be written

$$\langle HNF \rangle = ABC \cdots D \langle ENF \rangle D^{-1} \cdots C^{-1} B^{-1} A^{-1},$$

where A, B, C , etc., are the proper permutation transformations of the individual variables.

Summary

We have seen that E-space presents a richer structure than the more conventional D-space for expressing Boolean func-

tions. While E-space possesses an analogous geometrical structure, the N -cube, this structure has distinctly different properties. The major difference is the canceling of the intersection of parts which are on. This property results in an abundance of bases for the space. These bases are found to be fundamentally connected, through similarity transformations.

References

1. S. M. Reddy, "Easily Testable Realizations for Logic Functions," *IEEE Trans. Computers* C-21, 1183 (November 1972).
2. K. K. Saluja and S. M. Reddy, "Fault Detecting Test Sets For Reed-Muller Canonic Networks," *IEEE Trans. Computers* C-24, 995 (October 1975).
3. W. C. Carter, A. B. Wadia, and D. C. Jessep, Jr., "Implementation of Checkable Acyclic Automata by Morphic Boolean Functions," *Proceedings of the Symposium on Computers and Automata*, Polytechnic Institute of Brooklyn, 1971, p. 465.
4. John P. Robinson and Chia-Lung Yeh, "A Method for Modulo-2 Minimization," *IEEE Trans. Computers* C-31, 800 (August 1982).
5. G. Papakonstantiniou, "Minimization of Modulo-2 Sum of Products," *IEEE Trans. Computers* C-28, 163 (February 1979).
6. S. Even, I. Kohavi, and A. Paz, "On Minimal Modulo-2 Sums of Products for Switching Functions," *IEEE Trans. Electron. Computers* EC-16, 671 (October 1967).
7. M. Davio, J.-P. Deschamps, and A. Thayse, *Discrete and Switching Functions*, McGraw-Hill Book Co., Inc., New York, 1978.
8. I. S. Reed, "A Class of Multiple-Error-Correcting Codes and the Decoding Scheme," *Trans. Inst. Radio Engineers, Professional Group on Information Theory* PGIT-4, 38-49 (1954).
9. J. P. Roth, *Computer Logic, Testing and Verification*, Computer Science Press, Inc., Potomac, MD, 1980, Ch. 1.
10. W. Karnaugh, "The Map Method for Synthesis of Combinational Logic Circuits," *Communications and Electronics*, pp. 593-599 (1953).
11. F. F. Sellers, Jr., M. Y. Hsiao, and L. W. Bearnsen, *Error Detecting Logic For Digital Computers*, McGraw-Hill Book Co., Inc., New York, 1958, pp. 25-26.
12. D. E. Muller, "Application of Boolean Algebra To Switching Circuit Design and Error Detection," *Trans. Inst. Radio Engineers, Professional Group on Electronic Computers* PGEC-3, 6-12 (1954).

Received March 16, 1982; revised November 30, 1982

Harold Fleisher IBM Data Systems Division, P.O. Box 390, Poughkeepsie, New York 12602. Dr. Fleisher was appointed an IBM Fellow in 1974 in recognition of his contributions to optical data processing, computer systems, logic design and array logic. He joined IBM in 1950 as a member of the Advanced Development Laboratory, which became the Research Division. He received a B.A. in 1942 and an M.S. in 1943, both in physics and optics from the University of Rochester. In 1951 he received a Ph.D. in physics from Case Institute of Technology, now Case Western Reserve University. Dr. Fleisher was a Staff Member of the Radiation Laboratory, MIT, from 1943 through 1945, a senior engineer at the Rauland Corporation, Chicago, during 1946 and an instructor in physics at Case from 1946 to 1950. He received the OSRD certificate of merit for his work on radar while at MIT. He holds 37 patents and has received an Outstanding Invention Award from IBM. Dr.

Fleisher is a Fellow of the IEEE, a member of the American Physical Society, and past president of the local chapter of RESA/Sigma Xi. He was a member of the Admissions and Advancement Committee of the IEEE, Area Chairman, Eastern Region, for the IEEE Computer Society, and a member of the IEEE Publications Board and Curriculum Accreditation Committee of IEEE for Electrical Engineering and Computer Science. He is a Visiting Professor of Physics at Vassar College, and served as a Vassar Fellow in Vassar College's Science, Technology and Society Program.

Morton Tavel *Vassar College, Poughkeepsie, New York 12602.* Dr. Tavel is Professor of Physics and Director of the Program in Science, Technology and Society at Vassar College, where he has been since 1967. Previously, he was an Assistant Scientist in the Applied Mathematics division of Brookhaven National Laboratory from 1964 to 1967. He also spent a year (1971) as Visiting Associate Professor in the Department of Physics at Virginia Polytechnic Institute and a year (1977) as Visiting Associate Professor in the Department of Physics at SUNY Stony Brook. Dr. Tavel's field of specialization is mathematical physics, with emphasis on electromagnetic theory and transport phenomena. In the summers of 1981 and 1982 he worked with Dr. Harold Fleisher, IBM Fellow, Pough-

keepsie, New York, under the auspices of the IBM Summer Faculty Program on several problems involving the implementation of logic functions. Dr. Tavel obtained a B.S. in physics from the City College of New York in 1960, an M.S. in physics from Stevens Institute of Technology in 1962, and a Ph.D. in physics in 1964 from Yeshiva University, where he was a National Science Foundation doctoral fellow. He is a member of the American Physical Society, the American Association for the Advancement of Science, and Sigma Xi. Dr. Tavel is the author of two introductory monographs on electromagnetic theory and of many journal articles.

John David Yeager *Stanford University, Palo Alto, California 94305.* Mr. Yeager is a Ph.D. candidate in physics at Stanford University. He worked as a summer student at the IBM Data Systems Division in Poughkeepsie, New York, during the summers of 1981 and 1982. His work with IBM has been concerned with the development of algorithms associated with hardware design languages. In 1982, Mr. Yeager received his B.S. in physics and M.S. in mathematics from Case Western Reserve University in Cleveland, Ohio. He is a student member of the Institute of Electrical and Electronics Engineers and the American Physical Society.